

# Metode *Background Subtraction* untuk Deteksi Obyek Pejalan Kaki pada Lingkungan Statis

Achmad Solichin

Fakultas Teknologi Informasi, Universitas Budi Luhur  
Mahasiswa Jurusan IKE, FMIPA, Universitas Gadjah Mada  
Yogyakarta, Indonesia  
achmad.solichin@budiluhur.ac.id

Agus Harjoko

Jurusan Ilmu Komputer dan Elektronika, Fakultas MIPA,  
Universitas Gadjah Mada  
Yogyakarta, Indonesia  
aharjoko@ugm.ac.id

**Abstrak**—Penelitian ini mencoba menerapkan teknik pengurangan latar belakang (*background subtraction*) untuk melakukan segmentasi obyek pada citra. Citra digital diperoleh dari perekaman dengan menggunakan kamera genggam pada suatu area yang sering dilalui oleh pejalan kaki di dalam gedung. Obyek pejalan kaki dipisahkan dari citra latar belakang dengan melakukan pengurangan citra sederhana pada area RGB dan grayscale. Keberadaan *noise*, bayangan dan *ghost* dapat dihilangkan dengan proses penebalan (dilatasi) pada citra. Klasifikasi sederhana dilakukan berdasarkan bentuk dari obyek. Penelitian ini dapat mendeteksi obyek pejalan kaki dengan cukup baik hanya dengan menggunakan teknik seleksi berdasarkan ukuran dari obyek.

**Kata kunci**—*Background subtraction; pedestrian detection; lingkungan statis*

## I. PENDAHULUAN

Deteksi obyek pejalan kaki merupakan salah satu arah penelitian yang penting dalam rangka meningkatkan kemampuan sistem pengawasan di tempat-tempat umum. Markus Enzweiler dan Dariu M. Gavrilu dalam [1] menyimpulkan bahwa untuk mendeteksi obyek pejalan kaki di dalam citra dan video digital diawali dengan melakukan segmentasi citra awal atau disebut pemilihan ROI (*region of interest*). Salah satu teknik dasar untuk melakukan segmentasi awal terhadap citra digital adalah *sliding window* [2], [3] yang melakukan pencarian obyek pada keseluruhan citra berdasarkan citra *template* yang sudah ditentukan sebelumnya. Hal tersebut kurang efektif terutama jika diterapkan dalam sistem *real-time* karena membutuhkan waktu yang cukup lama. Beberapa teknik diusulkan untuk meningkatkan efektifitas segmentasi citra, misalnya dengan membagi komponen obyek terlebih dahulu seperti pada [4], [5] atau dengan menggunakan ciri yang didapatkan dari beberapa sumber citra sekaligus [6], [7].

Metode pemisahan obyek lainnya yang banyak digunakan adalah pengurangan citra latar belakang (*background subtraction*) [8], [9]. Pada dasarnya teknik pengurangan latar belakang ini cukup sederhana yaitu dengan melihat perbedaan antara citra tanpa obyek dengan citra berobyek. Metode ini sangat baik untuk mendeteksi pergerakan obyek dalam video [10], [11]. Dalam [12], dibandingkan beberapa metode

pengurangan citra latar belakang dilihat dari sisi kecepatan, penggunaan memori dan akurasi. Hasilnya metode *Mixture of Gaussian (MoG)* [13], [14] dan *Kernel Density Estimation (KDE)* [8], [15], [16] memiliki tingkat akurasi yang tinggi terutama pada situasi sederhana. Namun pada lingkungan yang kompleks dan luas, metode *Cooccurrence of image variations* [17] dan *Eigenbackgrounds* [18] lebih diunggulkan.

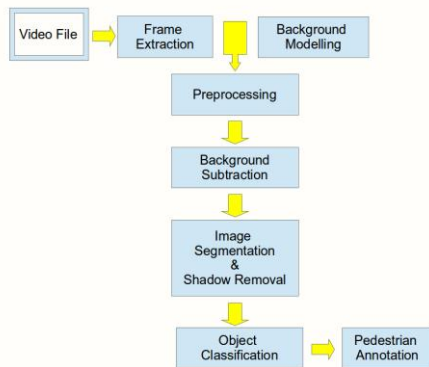
Salah satu permasalahan yang muncul pada penerapan metode pemisahan obyek menggunakan pengurangan latar belakang (*background subtraction*) adalah munculnya bayangan (*shadow*) dan *ghost*. Adanya bayangan yang ditimbulkan akibat cahaya yang mengenai pejalan kaki atau obyek lainnya akan terdeteksi sebagai obyek saat dilakukan pengurangan citra. Sedangkan *ghost* akan muncul akibat pergerakan kecil dari lingkungan seperti pergerakan daun karena tertiup angin atau akibat perubahan intensitas cahaya di dalam lingkungan. Bayangan dan *ghost* harus dihilangkan agar obyek pejalan kaki dapat tersegmentasi dengan sempurna. Untuk mendeteksi dan menghilangkan adanya *ghost* dapat dilakukan dengan membandingkan kemunculan obyek pada setiap *frame*. Jika obyek muncul tiba-tiba dan dalam jangka waktu yang singkat maka dapat disimpulkan obyek tersebut dikategorikan sebagai *ghost* [19]. Cara lainnya adalah dengan meningkatkan nilai batas saat melakukan segmentasi obyek, sehingga hanya obyek dengan ukuran dan intensitas tertentu yang akan tersegmentasi.

Teknik mendeteksi adanya bayangan dapat dibagi menjadi 4 (empat) kategori yaitu *statistical non-parametric*, *statistical parametric*, *deterministic non-model* berdasarkan warna dan *deterministic non-model* berdasarkan *spatial redundancy* [20]. Pada pendekatan *statistical non-parametric*, tidak memperhitungkan komponen warna dengan asumsi bahwa secara visual warna bayangan adalah sama namun memiliki kecerahan yang lebih rendah dibanding latar belakang. Secara lebih rinci, pendekatan ini dijelaskan pada [21], [22]. Sementara itu pada pendekatan *statistical parametric*, bayangan dideteksi berdasarkan informasi dari *pixel* dan domain ruang (*spatial*) dengan asumsi bahwa antara obyek dan bayangannya memiliki kedekatan lokasi. Contoh pendekatan *statistical parametric* dijelaskan pada [23]. Pendekatan *deterministic non-model* berdasarkan warna bekerja pada domain warna HSV karena dianggap paling mendekati persepsi

manusia [24]. Seperti sudah dijelaskan pada [25], pendekatan *deterministic non-model* berdasarkan domain spasial mendeteksi bayangan dengan mempertimbangkan adanya area yang lebih gelap, perbedaan cahaya yang mencolok dan adanya tepi yang tetap dan bergerak. Pendekatan terakhir ini dianggap paling baik dalam mendeteksi adanya bayangan.

Obyek yang sudah dipisahkan dari latar belakangnya dapat dilakukan proses klasifikasi untuk menentukan apakah obyek termasuk pejalan kaki atau bukan. Banyak pendekatan yang dapat dilakukan untuk mengklasifikasikan obyek pejalan kaki, antara lain berdasarkan bentuk (*shape*), tekstur dan bentuk, berdasarkan fitur lokal citra dan berdasarkan gabungan dari beberapa ciri pada citra [1]. Pada [26] dibandingkan 7 (tujuh) algoritma untuk deteksi pejalan kaki dan menyimpulkan *MultiFtr outperforms* [27] yang merupakan penggabungan dari teknik HOG [28] dan Haar [29] sebagai algoritma terbaik. Namun teknik deteksi pejalan kaki berdasarkan bentuk obyek tersegmentasi yang diusulkan pada [30] diklaim lebih efektif dan cepat. Lebih jauh lagi, penelitian pada [31] mengusulkan sistem deteksi pejalan kaki dengan menggunakan masukan dari beberapa kamera untuk menghasilkan ROI dan mengekstraksi fitur citra di dalamnya.

Penelitian ini menerapkan teknik pengurangan latar belakang (*background subtraction*) untuk melakukan segmentasi obyek pada citra. Citra digital diperoleh dari perekaman dengan menggunakan kamera tangan (*handycam*) pada suatu area yang sering dilalui oleh pejalan kaki di dalam gedung. Penelitian ini menganggap bahwa lingkungan bersifat statis dengan intensitas pencahayaan yang relatif sama. Keberadaan bayangan obyek dideteksi dan dihilangkan dari obyek sehingga hasil segmentasi obyek akan lebih baik. Hasil dari penelitian ini memberikan gambaran mengenai teknik deteksi obyek pejalan kaki pada lingkungan statis dengan menggunakan metode pengurangan citra latar belakang..



Gambar 1: Metode yang Diusulkan

## II. METODE YANG DIUSULKAN

Untuk memperoleh obyek pejalan kaki di dalam citra yang diperoleh dari video, dilakukan beberapa tahapan penelitian seperti digambarkan pada Gambar 1.

### A. Frame Extraction

Pada tahap awal file *video* ditangkap dan dipecah menjadi citra-citra digital berdasarkan *frame-frame video*. Proses ekstraksi dilakukan terhadap video dimana untuk setiap frame pada satuan waktu tertentu diubah ke dalam bentuk citra digital. Citra digital yang dihasilkan dalam bentuk JPEG (*Joint Photographic Experts Group*). Selanjutnya untuk setiap *pixel* di dalam citra digital dikonversi menjadi tipe *double* yang memiliki jangkauan nilai antara 0 hingga 1. Nilai *pixel* 0 untuk komponen warna yang lemah dan nilai 1 berarti komponen warna yang kuat. Walaupun diubah menjadi tipe *double*, namun citra digital masih berada dalam domain warna RGB (Red, Green, Blue).

### B. Background Modelling

Berdasarkan citra yang dihasilkan dari video, ditentukan model latar belakang. Selanjutnya dilakukan proses pemodelan citra latar belakang secara manual dengan memilih citra digital dimana tidak terdapat obyek di dalamnya. Pemodelan citra latar belakang ini dilakukan secara manual dengan batasan bahwa lingkungan bersifat statis sehingga citra latar belakang tanpa obyek juga bersifat statis. Dalam lingkungan statis, teknik ini lebih efektif dan cepat jika dibandingkan penerapan teknik pemodelan citra latar belakang adaptif [13].

### C. Preprocessing

Citra latar belakang dan citra yang diperoleh dilakukan normalisasi komponen R, G dan B. Normalisasi dilakukan untuk meningkatkan akurasi proses pengurangan citra. Untuk mendapatkan citra yang lebih baik, sebelum dilakukan pengurangan untuk setiap citra dan latar belakang dilakukan proses normalisasi komponen warna Red (R), Green (G) dan Blue (B). Komponen warna R, G dan B diambil dari setiap *pixel* di dalam citra, selanjutnya dilakukan normalisasi dengan persamaan (1), (2) dan (3).

$$R' = \frac{R}{R+G+B} \dots\dots\dots(1)$$

$$G' = \frac{G}{R+G+B} \dots\dots\dots(2)$$

$$B' = \frac{B}{R+G+B} \dots\dots\dots(3)$$

### D. Background Subtraction

Setelah citra dinormalisasi, selanjutnya dilakukan proses pengurangan citra (*background subtraction*) dengan cara melakukan pengurangan setiap *pixel* pada citra dengan obyek dengan citra latar belakang. Proses pengurangan citra dilakukan secara absolut dimana perbedaan setiap *pixel* dari kedua citra akan diperoleh dan selalu bernilai positif. Karena teknik ini akan melihat perbedaan untuk setiap *pixel* di dalam citra, sehingga kedua citra harus memiliki tipe data dan ukuran yang sama.

Sebagai contoh sederhana pengurangan citra digital secara absolut terlihat pada pengurangan citra X dan Y berikut ini

menjadi citra Z. Hasil pengurangan citra secara absolut akan selalu menghasilkan nilai positif.

```
X = uint8([ 255 10 75; 44 225 100]);
```

```
Y = uint8([ 50 50 50; 50 50 50 ]);
```

```
Z = imabsdiff(X,Y)
```

```
Z =
```

```
205  40  25
```

```
6 175  50
```

#### E. Image Segmentation dan Shadow Removal

Agar citra dapat tersegmentasi dengan baik, dilakukan beberapa proses penguatan (*dilation*) dan pengisian (*fill*) citra. Seperti sudah dijelaskan dalam [32], [33], penguatan citra A dengan citra B yang memiliki dimensi tertentu akan menghasilkan citra baru yang lebih tajam dan tebal.

Pada penelitian ini, pertama dilakukan penguatan (*dilation*) terhadap citra hasil pengurangan pada langkah sebelumnya. Proses penguatan citra dilakukan dengan matriks dimensi 9x9. Langkah kedua adalah melakukan pengurangan citra latar belakang dengan citra berobyek pada area *grayscale* (*gray level*). Kedua citra yang masih dalam bentuk RGB diubah menjadi citra *grayscale*. Setelah dihasilkan citra pengurangan dalam area *grayscale*, pada langkah ketiga kembali dilakukan penguatan (*dilation*) menggunakan matriks dimensi 3x3. Langkah keempat, dilakukan pengisian citra (*image fill*) untuk menghilangkan lubang-lubang (*holes*) setelah dilakukan proses penguatan. Langkah terakhir adalah menggabungkan kembali citra hasil pengurangan dalam area RGB yang sudah dilakukan proses penguatan dengan citra hasil pengurangan dalam area *grayscale*. Hasilnya adalah citra dengan obyek yang telah tersegmentasi dari latar belakangnya.

Sementara itu, proses penghilangan titik-titik *noise* dan bayangan yang mungkin muncul dilakukan dengan melakukan *filtering* terhadap citra hasil pengurangan dimana pixel-pixel yang bernilai di bawah 0.31 diturunkan menjadi 0 dan sebaliknya untuk pixel-pixel yang bernilai 0.31 ke atas diubah menjadi 1. Nilai 0.31 ini didapatkan berdasarkan hasil percobaan terhadap beberapa citra sedemikian hingga didapatkan nilai terbaik 0.31 tersebut.

#### F. Object Classification

Setelah obyek-obyek tersegmentasi dengan baik, dilakukan seleksi kembali dengan melakukan klasifikasi sehingga didapatkan obyek pejalan kaki dan obyek bukan pejalan kaki. Seleksi sederhana dilakukan dengan melihat ukuran setiap obyek. Ukuran obyek didapatkan dengan mendeteksi garis batas (*boundary*) untuk setiap obyek yang telah tersegmentasi. Setelah garis batas didapatkan, maka dilakukan perhitungan selisih terbesar dari setiap titik garis batas (*boundary*). Atau dengan kata lain, dilakukan perhitungan nilai diagonal terbesar dari setiap obyek. Berdasarkan hasil percobaan terhadap beberapa citra, didapatkan nilai selisih 100 pixel sedemikian hingga obyek dengan nilai selisih terbesar diatas 100 pixel

diklasifikasikan sebagai obyek pejalan kaki dan sebaliknya tidak diklasifikasikan sebagai obyek pejalan kaki.

#### G. Pedestrian Annotation

Setelah dilakukan klasifikasi obyek pejalan kaki dan bukan, obyek yang terklasifikasi sebagai pejalan kaki ditandai dengan kotak berwarna hijau. Kotak dibuat berdasarkan nilai garis batas (*boundary*) terluar dari setiap obyek. Dengan demikian obyek pejalan kaki akan terlihat jelas diantara obyek-obyek yang lain.

### III. HASIL DAN PEMBAHASAN

Berdasarkan metode penelitian yang telah diusulkan dan dijelaskan di atas, dilakukan serangkaian percobaan menggunakan perangkat lunak Matlab. Beberapa fungsi dari Matlab digunakan untuk mempermudah proses percobaan.

#### A. Frame Extraction

File video yang digunakan dalam percobaan ini didapatkan dengan melakukan perekaman menggunakan kamera tangan (*handycam*) pada salah satu sudut ruangan yang bersifat statis. File video berdurasi 1 menit 43 detik atau 103 detik. Dari file video tersebut, dilakukan ekstraksi citra untuk setiap 5 detik frame, sehingga dihasilkan sekitar 20 citra digital dalam format JPEG. Percobaan dilakukan terhadap 20 citra digital tersebut.

#### B. Background Modelling

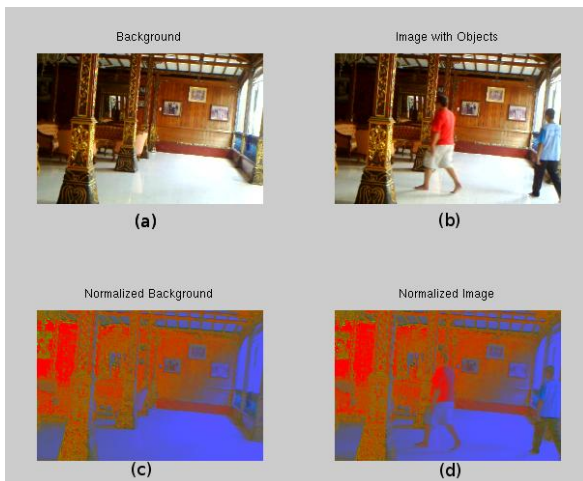
Berdasarkan citra digital yang telah diekstraksi dari file video, ditentukan model latar belakang. Penentuan model latar belakang dilakukan secara manual karena lingkungan bersifat statis. Gambar 2 merupakan citra model latar belakang yang ditentukan dalam penelitian ini.



Gambar 2: Citra Model Latar Belakang

#### C. Preprocessing

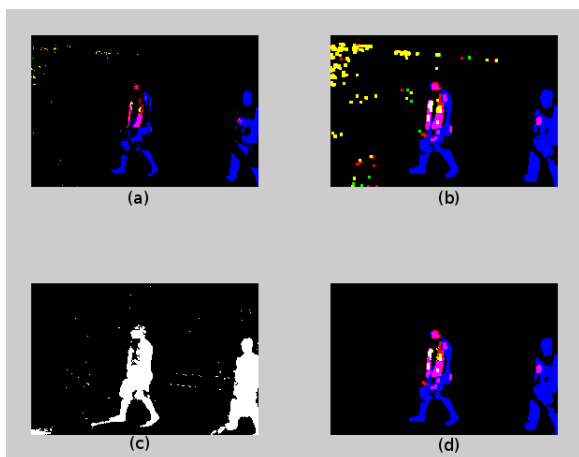
Citra latar belakang dan citra yang diperoleh dilakukan normalisasi komponen R, G dan B menggunakan persamaan (1), (2) dan (3). Citra hasil dari proses normalisasi terhadap Gambar 4 (a) dan (b) terlihat pada Gambar 4 (c) dan (d).



Gambar 3: Normalisasi Citra dan Latar Belakang

#### D. Background Subtraction

Setelah citra dinormalisasi, selanjutnya dilakukan proses pengurangan citra (background subtraction) dengan cara melakukan pengurangan setiap pixel pada citra dengan obyek dengan citra latar belakang secara absolut. Pada percobaan ini digunakan fungsi Matlab `imabsdiff(X, Y)` yang memiliki dua parameter masing-masing merupakan citra latar belakang (background) dan citra berobyek. Hasil dari fungsi ini adalah citra baru yang merupakan hasil pengurangan absolut antara kedua citra. Contoh hasil pengurangan citra dengan fungsi `imabsdiff()` yang dilakukan terhadap Gambar 3 (c) dan (d) adalah citra pada Gambar 4 (a).



Gambar 4: Proses Pengurangan Latar Belakang, Penghilangan Noise dan Bayangan

#### E. Image Segmentation & Shadow Removal

Pada proses segmentasi citra, pertama kali dilakukan penguatan (dilation) terhadap citra hasil pengurangan pada Gambar 4 (a). Proses penguatan citra dilakukan dengan menggunakan fungsi `imdilate()` yang sudah disediakan oleh Matlab. Penguatan citra dilakukan dengan matriks berdimensi

9x9. Hasil dari langkah ini terlihat pada Gambar 4 (b). Langkah kedua adalah melakukan pengurangan citra latar belakang dengan citra berobyek pada area grayscale (gray level). Kedua citra yang masih dalam bentuk RGB diubah menjadi citra grayscale dengan menggunakan fungsi `rgb2gray()`. Setelah itu, dilakukan pengurangan citra berobyek dengan citra latar belakang, dan hasilnya terlihat pada Gambar 4 (c). Langkah selanjutnya dilakukan penguatan (dilation) menggunakan matriks dimensi 3x3. Langkah keempat, dilakukan pengisian citra (image fill) dengan menggunakan fungsi `imfill()` untuk menghilangkan lubang-lubang (holes) setelah dilakukan proses penguatan. Langkah terakhir adalah menggabungkan kembali citra hasil pengurangan dalam area RGB yang sudah dilakukan proses penguatan dengan citra hasil pengurangan dalam area grayscale. Hasilnya adalah citra dengan obyek yang telah tersegmentasi dari latar belakangnya seperti terlihat pada Gambar 4 (d). Sementara itu, proses penghilangan titik-titik noise dan bayangan yang mungkin muncul dilakukan dengan melakukan filtering terhadap citra hasil pengurangan dimana pixel-pixel yang bernilai di bawah 0.31 diturunkan menjadi 0 dan sebaliknya untuk pixel-pixel yang bernilai 0.31 ke atas diubah menjadi 1.

#### F. Object Classification

Setelah obyek-obyek tersegmentasi dengan baik, dilakukan seleksi kembali dengan melakukan klasifikasi sehingga didapatkan obyek pejalan kaki dan obyek bukan pejalan kaki. Seleksi sederhana dilakukan dengan melihat ukuran setiap obyek yang didapatkan dengan mendeteksi garis batas (boundary) untuk setiap obyek yang telah tersegmentasi. Pendeteksian garis batas untuk setiap obyek memanfaatkan fungsi `bwboundaries()` yang sudah disediakan oleh Matlab. Gambar 5 menunjukkan garis batas (boundary) berwarna merah yang mengelilingi obyek yang sudah tersegmentasi. Setelah garis batas didapatkan, maka dilakukan perhitungan selisih terbesar dari setiap titik garis batas (boundary). Nilai terbesar didapatkan dengan menggunakan fungsi `max()` yang diterapkan terhadap setiap boundary, sedangkan nilai terkecil didapatkan dengan fungsi `min()`. Dengan menggunakan nilai selisih minimal 100 pixel maka dapat dipisahkan obyek pejalan kaki dengan obyek bukan pejalan kaki.

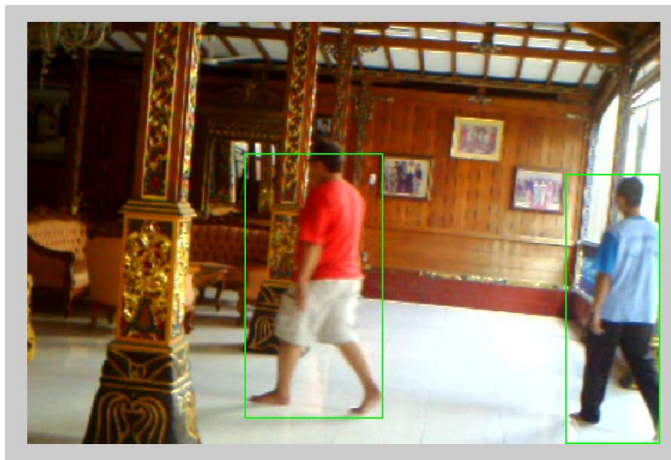


Gambar 5: Image Boundaries



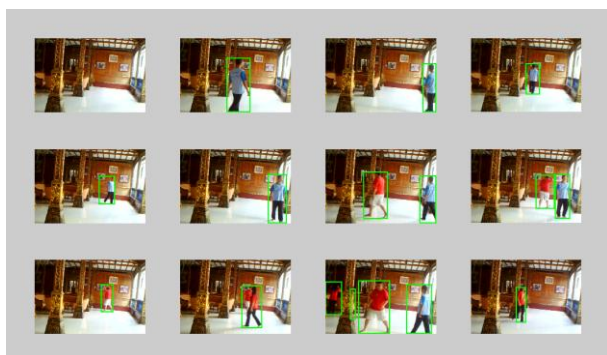
### G. Pedestrian Annotation

Setelah dilakukan klasifikasi obyek pejalan kaki dan bukan, obyek yang terklasifikasi sebagai pejalan kaki ditandai dengan kotak berwarna hijau. Pembuatan kotak tersebut dilakukan dengan fungsi `rectangle()` dari Matlab. Hasilnya terlihat pada Gambar 6.



Gambar 6: Penandaan Obyek Pejalan Kaki (Pedestrian Annotation)

Sementara itu, pada Gambar 7 terlihat hasil deteksi pejalan kaki untuk beberapa citra digital yang diperoleh dari video.



Gambar 7: Hasil Deteksi Pejalan Kaki untuk beberapa Citra

## IV. KESIMPULAN

Berdasarkan hasil penelitian dan percobaan yang telah dilakukan, dapat disimpulkan bahwa teknik pengurangan latar belakang (*background subtraction*) menghasilkan proses deteksi yang cukup baik. Pada lingkungan statis dengan lokasi di dalam ruangan dimana intensitas pencahayaan yang relatif tetap, latar belakang yang digunakan dapat dimodelkan secara manual. Namun pada lingkungan dengan kondisi yang dinamis, diperlukan latar belakang yang adaptif terhadap kondisi lingkungan. Penelitian ini juga dapat mendeteksi obyek pejalan kaki dengan cukup baik hanya dengan menggunakan teknik seleksi berdasarkan ukuran dari obyek. Untuk meningkatkan akurasi, dapat dilakukan teknik perbandingan dengan data training yang disiapkan sebelumnya. Selain itu, berdasarkan hasil uji coba yang telah dilakukan, metode yang diusulkan

pada penelitian ini sangat dipengaruhi oleh intensitas cahaya ruangan. Pada intensitas cahaya yang berbeda antara citra latar dengan citra berobyek, proses deteksi pejalan kaki tidak menunjukkan hasil yang baik.

## DAFTAR PUSTAKA

- [1] M. Enzweiler and D. M. Gavrila, "Monocular pedestrian detection: survey and experiments,," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 12, pp. 2179–2195, Dec. 2009.
- [2] M. A. Turk and A. P. Pentland, "Face Recognition Using Eigenfaces,," in Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1991, 1991, pp. 586–591.
- [3] C. H. Lampert, M. B. Blaschko, and T. Hofmann, "Beyond sliding windows: Object localization by efficient subwindow search,," in IEEE Conference on Computer Vision and Pattern Recognition, 2008, pp. 1–8.
- [4] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-Based Object Detection in Images by Components,," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 4, pp. 349–361, 2001.
- [5] H. Schneiderman and T. Kanade, "Object Detection Using the Statistics of Parts,," vol. 56, no. 3, pp. 151–177, 2004.
- [6] M. Enzweiler, A. Eigenstetter, B. Schiele, and D. M. Gavrila, "Multi-cue pedestrian classification with partial occlusion handling,," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, 2010, pp. 990–997.
- [7] S. Munder, C. Schnorr, and D. M. Gavrila, "Pedestrian Detection and Tracking Using a Mixture of View-Based Shape-Texture Models,," IEEE Transactions on Intelligent Transportation Systems, pp. 1–25, 2008.
- [8] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction,," in IEEE FRAME-RATE WORKSHOP, 2000, pp. 751–767.
- [9] S. S. Cheung and C. Kamath, "Robust techniques for background subtraction in urban traffic video,," Visual Communications and Image Processing, vol. 5308, pp. 881–892, 2004.
- [10] P. Spagnolo, T. D. Orazio, M. Leo, and a. Distanto, "Moving object segmentation by background subtraction and temporal analysis,," Image and Vision Computing, vol. 24, no. 5, pp. 411–423, May 2006.
- [11] C. Peijiang, "Moving Object Detection Based on Background Extraction,," in International Symposium on Computer Network and Multimedia Technology, 2009, pp. 1–4.
- [12] M. Piccardi, "Background subtraction techniques: a review,," in 2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583), 2004, pp. 3099–3104.
- [13] C. Stauffer and W. E. L. E. L. Grimson, "Adaptive background mixture models for real-time tracking,," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1999, pp. 246–252.
- [14] Z. Chen and T. Ellis, "Self-adaptive Gaussian mixture model for urban traffic monitoring system,," 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 1769–1776, Nov. 2011.
- [15] A. Mittal, N. Paragios, and C. Marne, "Motion-Based Background Subtraction using Adaptive Kernel Density Estimation,,"
- [16] Elgammal, R. Duraiswami, D. Harwood, and L. S. S. Davis, "Background and foreground modeling using nonparametric kernel density estimation for visual surveillance,," Proceedings of the IEEE, vol. 90, no. 7, pp. 1151–1163, Jul. 2002.
- [17] M. Seki, T. Wada, H. Fujiwara, and K. Sumi, "Background subtraction based on cooccurrence of image variations,," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003, 2003, vol. 2, pp. 65–72.
- [18] N. M. Oliver, B. Rosario, and A. P. Pentland, "A Bayesian computer vision system for modeling human interactions,," IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pp. 831–843, 2000.

- [19] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting Moving Objects, Ghosts, and Shadows in Video Streams," *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, vol. 25, no. 10, pp. 1337–1342, Oct. 2003.
- [20] A. Prati, R. Cucchiara, I. Miki, and M. M. Trivedi, "Analysis and Detection of Shadows in Video Streams : A Comparative Evaluation," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001., 2001, vol. 00, no. C, pp. II-571 – II-576 vol.2.
- [21] I. Haritaoglu, D. Harwood, and L. S. S. Davis, "W4: Real-time Surveillance of People and Their Activities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 809–830, 2000.
- [22] T. Horprasert, D. Harwood, and L. S. Davis, "A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection," in *Horprasert99 a statistical*, 1999, pp. 1–19.
- [23] I. Mikic, P. C. C. Cosman, G. T. T. Kogut, and M. M. M. Trivedi, "Moving shadow and object detection in traffic scenes," in *Proceedings 15th International Conference on Pattern Recognition*, 2000, vol. 1, pp. 321–324.
- [24] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting Objects, Shadows and Ghosts in Video Streams by Exploiting Color and Motion Information," in *Proc. 11th International Conference on Image Analysis and Processing*, 2001., 2001, pp. 360–365.
- [25] J. Stauder, R. Mech, and J. Ostermann, "Detection of Moving Cast Shadows for Object Segmentation," *IEEE Transactions on Multimedia*, vol. 1, pp. 65–76, 1999.
- [26] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian Detection : A Benchmark," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009. CVPR 2009., 2009, pp. 304–311.
- [27] C. Wojek and B. Schiele, "A Performance Evaluation of Single and Multi-feature People Detection," in *Proceedings of the 30th DAGM symposium on Pattern Recognition*, 2008, pp. 82–91.
- [28] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005., 2005, pp. 1–13.
- [29] C. Papageorgiou and T. Poggio, "A Trainable System for Object Detection," *International Journal of Computer Vision*, vol. 38, no. 1, pp. 15–33, 2000.
- [30] Y. Bo and C. C. Fowlkes, "Shape-based pedestrian parsing," in *Cvpr* 2011, 2011, pp. 2265–2272.
- [31] C. G. Keller, M.ENZWEILER, and M. Rohrbach, "The benefits of dense stereo for pedestrian detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1096 – 1106, 2011.
- [32] R. C. Gonzalez and R. E. Woods, *Digital Image Processing Third Edition*. 2010, pp. 655–657.
- [33] R. van den Boomgard and R. van Balen, "Methods for Fast Morphological Image Transforms Using Bitmapped Images," *Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing*, vol. 54, no. 3, pp. 254–258, 1992.